

CLAIMS

Please amend claims 1, 4, 15, 17, and 37 as indicated:

1. (Currently Amended) A computer program product recorded on a non-transitory computer readable medium for organizing and manipulating Web services software modules in containers on a network, comprising:

computer executable instructions to determine and describe Web services software modules in containers that are available at a corresponding, local network node, said Web services software modules comprising executable software modules that can be exchanged between nodes of a network and run at said nodes;

computer executable instructions to generate messages to be transmitted to other containers via a network disclosing said Web services software modules that are available at said corresponding network node, and including contextual information about said containers and said Web services available at said corresponding, local node;

computer executable instructions to receive and decipher messages disclosing Web services software modules that are available at other network nodes corresponding to other containers; and

computer executable instructions to cause the dynamic reconfiguration of said Web services software modules available at said corresponding network node on said network based on said transmitted and received messages, including the exchange of said Web services software modules between said network nodes, wherein said instructions to cause the dynamic reconfiguration comprise:

computer executable instructions to transmit messages that are hardware and software platform independent to said other containers, requesting said other containers to return copies of Web services software modules;

computer executable instructions, responsive to receipt of messages from said other containers requesting copies of Web services software modules available at said corresponding network node, to send copies of said requested Web services software modules to said requesting containers;

computer executable instructions to receive client requests for use of a Web services software module from client computers via said network;

computer executable instructions that, responsive to receipt of one of said client requests from a client for a Web services software module that is not available at said corresponding network node, determine[[s]], based on said received messages disclosing

said Web services software modules that are available at other network nodes, whether another network node has a copy of said particular Web services software module; and
 computer executable instructions that invoke a proxy to another of said containers having a copy of a particular Web services software module based on said determination.

2-3. (Cancelled)

4. (Currently amended) The computer program product of Claim 1 wherein said computer executable instructions to transmit messages further comprises computer executable instructions to transmit said messages to and from a Web services registry and said computer executable instructions to receive and decipher messages further comprises computer executable instructions to receive said messages from a Web service registry.
5. (Previously Presented) The computer program product of Claim 4 wherein said messages disclosing said Web services software modules that are available at network nodes are in the Web Services Descriptor Language (WSDL).
6. (Previously Presented) The computer program product of Claim 5 wherein said registry is a Universal Description, Discovery, and Integration initiative (UDDI) registry.
7. (Previously Presented) The computer program product of Claim 1 wherein said computer executable instructions to transmit messages uses a peer to peer messaging protocol between said containers and said computer executable instructions to receive and decipher messages uses a peer to peer messaging protocol between containers.
8. (Previously Presented) The computer program product of Claim 7 wherein said messaging protocol is SOAP.
9. (Previously Presented) The computer program product of Claim 8 wherein said disclosures of said Web services software modules that are available at network nodes are contained in headers of Simple Object Access Protocol (SOAP) messages.
10. (Previously Presented) The computer program product of Claim 7 wherein said messaging

protocol is JXTA.

11-12. (Cancelled)

13. (Previously Presented) The computer program product of claim 1 wherein said proxy comprises:

computer executable instructions to route said client requests for a Web services software module that is not available at said corresponding network node and has been determined to be available at another network node to another container corresponding to said another network node;

computer executable instructions to receive responses to said client requests from said another network node; and

computer executable instructions to return said responses to said requesting clients.

14. (Previously Presented) The computer program product of claim 13 further comprising:

computer executable instructions to receive said client requests routed from another of said containers and causing said client requests to be handled by a copy of said particular Web services software module at a network node corresponding to said container to generate said response; and

computer executable instructions to transmit said response to said another container that routed said client request to said container.

15. (Currently Amended) The computer program product of claim 1 further comprising:

computer executable instructions to determine a load of client requests at said corresponding network node; and

wherein said computer executable instructions ~~for causing to cause~~ the dynamic reconfiguration of Web services software modules performs said dynamic reconfiguration based on said load determination.

16. (Previously Presented) The computer program product of claim 15 wherein said computer executable instructions to cause the dynamic reconfiguration of Web services software modules further comprises:

computer executable instructions that, responsive to determination of a load of client requests for a particular Web services software module that is not available at said corresponding network node exceeding a predetermined level, issues a message requesting a copy of said particular Web services software module from another container that has a copy of said particular Web services software module;

computer executable instructions to receive and locally invoke said particular Web services software module from said other container; and

computer executable instructions to route client requests for said particular Web services software module to said local invocation of said particular Web services software modules.

17. (Currently Amended) The computer program product of claim 16 wherein said computer executable instructions to cause the dynamic reconfiguration of Web services software modules further comprises:

computer executable instructions ~~for offloading to offload~~ said particular Web services software module received from said other container responsive to said load of client requests for said particular Web services software module dropping below a second predetermined level.

18. (Previously Presented) The computer program product of claim 15 wherein said computer executable instructions to cause the dynamic reconfiguration of Web services software modules comprises:

computer executable instructions that, responsive to determination of a load of client requests for a particular Web services software module available at said corresponding network node exceeding a predetermined level, issues a message requesting another container to accept a copy of the code of said particular Web services software modules from said computer program product; and

computer executable instructions to send a copy of said code of said particular Web services software module to said other container responsive to affirmative responses to said message requesting another container to accept a copy of the code of said particular Web services software module from said computer program product.

19. (Previously Presented) The computer program product of claim 18 wherein said computer executable instructions to cause the dynamic reconfiguration of Web services software modules further comprises:

computer executable instructions to reconfigure said computer program product to route client requests for said particular Web services software module to said other container.

20. (Previously Presented) The computer program product of claim 19 wherein said other container comprises a plurality of other containers.

21. (Previously Presented) The computer program product of claim 20 wherein said computer executable instructions to reconfigure said computer program product to route client requests for said particular Web services software module to said other container distributes said client requests for said particular Web services software module between said other containers and said local invocation of said particular Web services software module.

22. (Previously Presented) The computer program product of claim 1 wherein said client requests indicate whether said requesting client has a container and a platform on which said client is running and wherein said computer program product further comprises computer executable instructions to read said client requests to determine whether said client has a container and said platform.

23. (Previously Presented) The computer program product of claim 22 wherein said computer executable instructions to cause the dynamic reconfiguration of Web services software modules further comprising:

computer executable instructions to send a copy of the code of a particular Web services software module responsive to a client request for said Web services software module.

24. (Previously Presented) The computer program product of claim 1 further comprising:

computer executable instructions to monitor usage of Web services software modules by clients; and

computer executable instructions to charge said clients for said usage.

25. (Previously Presented) A method for organizing and manipulating Web services software modules in containers, comprising:

determining and describing Web services software modules in containers that are

available at a corresponding network node, said Web services software modules comprising executable software modules that can be exchanged between nodes of a network and run at said nodes;

transmitting messages via a network disclosing said Web services software modules that are available at said corresponding network node to other network nodes via said network and including contextual information about said container and said Web services available at said corresponding, local node;

receiving and deciphering messages from other network nodes disclosing Web services software modules that are available at other network nodes; and

dynamically reconfiguring Web services software modules on said network node transmitted and received based on said messages, including the exchange of said Web services software modules between said network nodes , wherein said instructions for causing the dynamic reconfiguration comprise:

transmitting messages that are hardware and software platform independent to said other containers, requesting said other containers to return copies of Web services software modules;

responsive to receipt of messages from said other containers, requesting copies of Web services software modules available at said corresponding network node, for sending copies of said requested Web services software modules to said requesting containers;

receiving client requests for use of a Web services software module from client computers via said network;

responsive to receipt of a client request from a client for a Web services software module that is not available at said corresponding network node, determining, based on said received messages disclosing said Web services software modules that are available at other network nodes, what network nodes have copies of said particular Web services software module; and

invoking a proxy to another of said network nodes having a copy of a particular Web services software module based on said determination.

26-27. (Cancelled)

28. (Previously Presented) The method of Claim 25 wherein transmitting and receiving messages comprises sending and receiving said messages to and from a Web services registry.

29. (Previously Presented) The method of Claim 28 wherein said messages disclosing said Web services software modules that are available at network nodes are in the Web Services Descriptor Language (WSDL).

30. (Previously Presented) The method of Claim 29 wherein said registry is a Universal Description, Discovery, and Integration initiative (UDDI) registry.

31. (Previously Presented) The method of Claim 25 comprising sending and receiving said messages using a peer to peer messaging protocol between said network nodes.

32. (Previously Presented) The method of Claim 31 wherein said messaging protocol is SOAP.

33. (Previously Presented) The method of Claim 32 wherein said disclosures of said Web services software modules that are available at network nodes are contained in headers of Simple Object Access Protocol (SOAP) messages.

34. (Previously Presented) The method of Claim 31 wherein said messaging protocol is JXTA.

35-36. (Cancelled)

37. (Currently Amended) The method of claim 25 wherein said proxy routes client requests for said particular Web services software module to said other of said network nodes, receives responses to said client requests, and ~~returning~~ returns said responses to said requesting clients.

38. (Previously Presented) The method of claim 37 further comprising:

receiving said client requests forwarded from other of said network nodes and causing said client requests to be handled by said copy of said particular Web services software module corresponding to said network node to generate said response; and

transmitting said response to said network node that issued said client request.

39. (Previously Presented) The method of claim 25 further comprising:
- determining a load of client requests at said corresponding network node; and
wherein-said dynamic reconfiguration is performed based on said load determination.
40. (Previously Presented) The method of claim 39 wherein dynamically reconfiguring further comprises:
- responsive to determination of a load of client requests for a particular Web services software module that is not available at said corresponding network node exceeding a predetermined level, issuing a message requesting a copy of the code of said particular Web services software module from another network node that has a copy of said particular Web services software module;
- receiving and locally invoking said code for said particular Web services software module from said other network node; and
- routing client requests for said particular Web services software module to said local invocation of said code for said particular Web services software module.
41. (Previously Presented) The method of claim 40 wherein dynamically reconfiguring further comprises:
- offloading said local code for said particular Web services software module responsive to said load of client requests for said particular Web services software module dropping below a second predetermined level.
42. (Previously Presented) The method of claim 39 wherein dynamically reconfiguring comprises:
- responsive to determination of a load of client requests for a particular Web services software module available at said corresponding network node exceeding a predetermined level, issuing a message requesting another network node to accept a copy of the code of said particular Web services software module from said network node; and
- sending a copy of said code of said particular Web services software module to said other network node responsive to affirmative responses to said message requesting another network node to accept a copy of the code of said particular Web services software module from said network node.

43. (Previously Presented) The method of claim 42 wherein dynamically reconfiguring further comprises:

reconfiguring said network node to route client requests for said particular Web services software module to said other network node.

44. (Previously Presented) The method of claim 43 wherein said other network node comprises a plurality of other network nodes.

45. (Previously Presented) The method of claim 44 wherein reconfiguring said network node comprises distributing said client requests for said particular Web services software module between said other network nodes and said local invocation of said particular Web services software module.

46. (Previously Presented) The method of claim 25 wherein said client requests indicate a platform on which said client is running and wherein said method further comprises:

reading said client requests to determine said platform of said client.

47. (Previously Presented) The method of claim 46 further comprising:

sending a copy of the code of a particular Web services software module responsive to a client request for said Web services software module.

48. (Previously Presented) The method of claim 25 further comprising:

monitoring usage of Web services software modules by clients; and
charging said clients for said usage.

49. (Previously Presented) The computer program product of claim 1 wherein said contextual information includes at least one of an identity of a Web service, the capabilities of said Web service, the operating system of said Web service, the platform of said Web service, the Web services hosted by a container type Web service, the workload of said Web service, and a network location of said Web service.